

(12)特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局



(43) 国際公開日
2003年5月15日 (15.05.2003)

PCT

(10) 国際公開番号
WO 03/040948 A1

(51) 国際特許分類⁷:

G06F 15/177

(72) 発明者; および

(21) 国際出願番号:

PCT/JP01/09788

(75) 発明者/出願人(米国についてのみ): 成瀬
彰 (NARUSE, Akira) [JP/JP]; 久門耕一 (KUMON,
Kouichi) [JP/JP]; 佐藤 充 (SATO, Mitsuru) [JP/JP]; 〒
211-8588 神奈川県川崎市中原区上小田中4丁目1番1
号 富士通株式会社内 Kanagawa (JP).

(22) 国際出願日: 2001年11月8日 (08.11.2001)

(25) 国際出願の言語:

日本語

(74) 代理人: 竹内 進 (TAKEUCHI, Susumu); 〒105-0003
東京都港区西新橋3丁目25番47号 清水ビル8階 Tokyo
(JP).

(26) 国際公開の言語:

日本語

(81) 指定国(国内): JP, US.

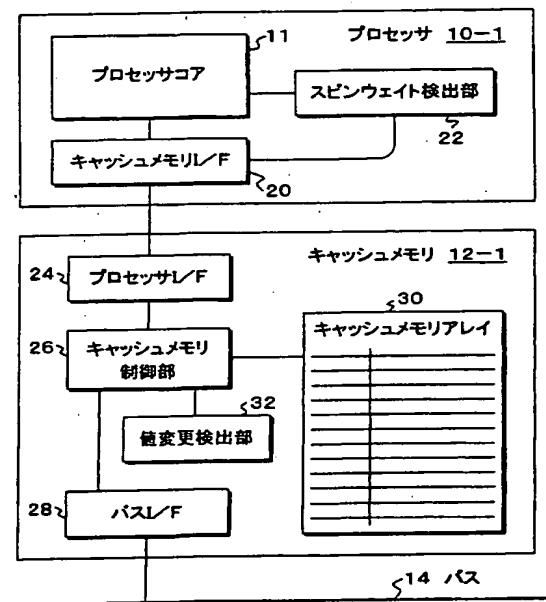
(71) 出願人(米国を除く全ての指定国について): 富士通
株式会社 (FUJITSU LIMITED) [JP/JP]; 〒211-8588 神
奈川県川崎市中原区上小田中4丁目1番1号 Kanagawa
(JP).

添付公開書類:
— 国際調査報告書

[統葉有]

(54) Title: COMPUTER AND CONTROL METHOD

(54) 発明の名称: コンピュータ及び制御方法



10-1...PROCESSOR
11...PROCESSOR CORE
20...CACHE MEMORY IF
22...SPIN WEIGHT DETECTING UNIT
12-1...CACHE MEMORY
24...PROCESSOR IF
26...CACHE MEMORY CONTROL UNIT
30...CACHE MEMORY ARRAY
32...VALUE CHANGE DETECTING UNIT
28...BUS IF
14...BUS

WO 03/040948 A1

(57) Abstract: A computer including a plurality of processors having a cache memory. A spin weight detecting unit of the processor gives an instruction to monitor a variable which is a condition to end a spin weight when it detects execution of a spin weight instruction, and then changes the operating state of the processor. A value change detecting unit of the cache memory monitors the variable specified by the spin weight detecting unit and notifies the processor of the value change when it detects that the variable has been changed to return the operating state to the previous one.

[統葉有]



2文字コード及び他の略語については、定期発行される各PCTガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(57) 要約:

キャッシュメモリを備えた複数のプロセッサを有するコンピュータである。プロセッサに設けたスピニウェイト検出部は、スピニウェイト命令の実行中を検出した際に、スピニウェイトの終了条件となっている変数値の監視を指示した後にプロセッサの動作状態を変更させる。キャッシュメモリに設けた値変更検出部は、スピニウェイト検出部から指示された変数値を監視し、この変数値が変更されたことを検出した場合にプロセッサに値変更を通知して動作状態を元に戻す。

明細書

コンピュータ及び制御方法

5 技術分野

本発明は、キャッシュメモリを備えた複数のプロセッサを有する共有メモリ型マルチプロセッサ構成のコンピュータ及び制御方法に関し、特に、同期処理などに用いられるスピンドルウェイト命令の命令実行効率を改善するためのコンピュータ及び制御方法に関する。

10

背景技術

従来、キャッシュメモリを備えた複数のプロセッサでメインメモリを共有する共有メモリ型マルチプロセッサシステムにおいては、プロセッサ間の同期処理にスピンドルウェイトが用いられることが多い。

15 このプロセッサ間の同期処理にスピンドルウェイトを使用する代表的な例がスピンドルロックである。スピンドルロックの命令処理は図1のように、検証部100と設定部102の手順で実現されることが多い。検証部100は、ロック獲得ポイントに達したプロセス又はスレッドがステップS1で変数Xをロードし、ステップS2で変数Xが例えばX=0といったロック獲得が可能であることを示す値か否か判定し、X=0になるまでステップS1、S2の処理を繰り返すスピンドルを行って待つスピンドルウェイトを行う。設定部102は、「test_and_set」や「compare_and_jump」などのインターロック命令を用いて、ステップS3でロック変数値XをX=1に設定してスピンドルロック処理を抜ける。もしロック変数値X=1の設定に失敗したら検証部100に戻りスピンドルウェイトを継続する。このように、ロック獲得が可能であるかどうかをチェックするスピンドルロックの検証部100でスピンドルウェイトが用いられることが多い。

20

25

スピンドルロック以外では、共有メモリ型マルチプロセッサシステムにおけるプロセッサ間の同期処理にスピンドルウェイトを使用するバリア同期がある。バリア同期とは、同期対象であるプロセス又はスレッドが同期ポイントで待ち合わせを行う

ことであり、メモリベースのバリア同期の場合には、図2の手順で実現されることが多い。まず設定部104のステップS1において、プロセス又はスレッドが同期ポイントに達する前はロック変数XはX=0である。同期ポイントに達するとプロセス又はスレッドは、「test_and_set」や「compare_and_jump」などのインターロック命令を用いて該当するプロセス又はスレッドに対応するロック変数Xのビットを1に設定し、検証部106に進む。検証部106は、ステップS2でロック変数Xをロードし、ステップS3でバリア同期の対象となっている全プロセス又は全スレッドのビットが1になるまで、即ち all X=1までスピンして待つスピンウェイトを行う。全プロセス又は全スレッドが同期ポイントに達すると、all X=1になり、同期がとられた状態でスピンウェイトを抜けて次の処理に移行する。このようにバリア同期でも、全プロセス又は全スレッドが同期ポイントに達しているかどうかをチェックする検証部106でスピンウェイトが用いられることが多い。

また、I/Oデバイス（入出力デバイス）との同期にもスピンウェイトが使用されている。通常のI/O処理では、プロセッサとI/Oデバイス間の同期には割り込みを使うのが一般的であった。これは、I/Oデバイスがプロセッサと比べて非常に遅く、I/Oデバイスからの応答をプロセッサが待ち続けることの損失が大きかったためである。しかし、I/Oデバイスの高速化により、一部のI/Oデバイスにおいては割り込みを使うことによる弊害が出てきている。一般的に割り込みはオーバーヘッド（遅延）が大きいので、I/Oデバイス本来の速度を活かせない。そのため、一部の高速I/Oデバイスに関しては、スピンウェイトによる同期処理を採用し始めている。

しかしながら、スピンウェイトの実行には無駄が多い。スピンウェイトは、ウェイト終了条件となっている変数値が他のプロセッサやI/Oデバイスなどの他のエージェントにより所望の値に変更されるまで延々と同じ命令列の実行を繰り返す。繰り返しの回数となるスピン回数は、場合によっては数百回、数千回に及ぶこともめずらしくなく、ある意味でプロセッサを無駄に使用しているといえる。電力消費の観点から考えると、スピンウェイトのあいだは無駄に電力を消費していることになる。

また、複数スレッドの同時実行が可能なSMT (Simultaneous Multi-Threading architecture) プロセッサにおいては、スピンウェイトを実行中のスレッドが他の実行スレッドの妨げとなる可能性がある。一般的にSMTプロセッサでは、プロセッサリソースはスレッド間で共有されている。このプロセッサリソースは、実行スレッド毎に均等に割当てられるのではなく、各スレッドの状態に応じて割当てられるプロセッサリソースの数や量が変わるというのが一般的である。プロセッサリソース割当てを決定する基準にはいろいろな方法が提案されているが、例えば次のようなアプローチがある。

(1) キャッシュミスを起こしたスレッドへのリソース割当てを減らす；

10 (2) 投機的に実行している命令数が多いスレッドへのリソース割当てを減らす；

(3) リザベーションステーションに登録されている命令数の多いスレッドへのリソース割当てを減らす；

15 これらのアプローチは、よりスムーズに命令を実行できる可能性の高いスレッドに優先的にプロセッサリソースを割当てようという考えに基づいている。ここスピニウェイトは命令を非常にスムーズに実行することができる。なぜなら、同じ命令列の実行を繰り返しているだけなので、キャッシュミスも起らないし、分岐予測も外れることがないからである。したがって、SMTプロセッサにおいては、スピニウェイトを実行中のスレッドに期せずして優先的にプロセッサリソースを割当ることになる。すると、スピニウェイト実行中スレッドに比べ、生産性の高い命令列を実行していると思われる他のスレッドへのプロセッサリソースの割当てが減ってしまい、結果としてプロセッサの性能が下がる可能性がある。

20 この様に、スピニウェイトには

(1) 電力を無駄に消費する；

25 (2) SMTプロセッサの性能が下げる可能性が高い；

といった問題がある。

この問題への対策として、スピニウェイトが始まつたことを検出したら、スピニウェイトの実行を停止し、スピニウェイトから抜けられる条件が整つたら割り込みにより通知し、実行を再開するという手法もある。しかし、スピニウェイト

から抜けられる条件が整つたことを、割り込みで通知してしまっては元も子もない。そもそもスピニウェイトは低遅延を実現するために採用しているのであり、割り込みを使つたら時間コストが大きくなってしまう。

また、同期専用のハードウェアを備えるという手法もある。しかし、その場合 5 にはハードウェアコストが高くなる。現在、メモリベースの同期処理が一般的となつてゐる背景には、汎用デバイス(メモリ)を用いて同期処理を行うことによるコスト削減のメリットが大きいという歴史的な経緯がある。

本発明は、スピニウェイトの実行で生じる電力やプロセッサリソースの無駄を省いて命令実行効率を高めるコンピュータ及び制御方法を提供することを目的と 10 する。

発明の開示

本発明は、キャッシュメモリを備えた複数のプロセッサを有するコンピュータにおいて、プロセッサによるスピニウェイト命令の実行中を検出した際に、スピニウェイトの終了条件となつてゐる変数値の監視を指示した後にプロセッサの動作状態を変更させるスピニウェイト検出部と、スピニウェイト検出部から指示された変数値を監視し、この変数値が変更されたことを検出した場合にプロセッサに値変更を通知して動作状態を元に戻す値変更検出部とを設けたことを特徴とする。このように本発明は、スピニウェイトから脱出できる可能性がない時、もしくは可能性が低い時は、プロセッサの動作状態を変更することにより、スピニウェイト実行中に生ずる無駄を削減する。また本発明によれば、プロセッサにスピニウェイト検出部を設け、またキャッシュメモリに値変更検出部を設けるといった少量のハードウェアの追加により、スピニウェイトが終了する可能性が無いか又は低いときの無駄を削減し、スピニウェイトの低遅延という特徴を損ねることなく、プロセッサの命令実行効率を良くする。

スピニウェイト検出部は、プロセッサによるスピニウェイト命令の実行中を検出した際に、スピニウェイト命令の実行で参照している変数値が格納されたキャッシュメモリのキャッシュブロックの監視を値変更検出部に指示する。値変更検出部は、キャッシュブロックの状態が他のプロセッサ又は入出力ユニットにより

無効化されたことを検出した場合に、プロセッサに値変更を通知する。変更検出部は、キャッシングブロックがキャッシングメモリから追い出されたことを検出した場合にも、プロセッサに値変更を通知する。

5 スピンウェイト検出部は、プロセッサによるスピンウェイト命令の実行中を検出した際に、スピンウェイト命令の実行で参照している変数値が格納されているメモリアドレスに対するバスリクエストの監視を値変更検出部に指示する。値変更検出部は、スピンウェイト検出部から指示された変数値が格納されているメモリアドレスに対するバスリクエストを監視し、他のプロセッサ又は入出力ユニットから前記メモリアドレスに対するバスリクエストを検出した場合に、プロセッサに値変更を通知する。

10 プロセッサは通常モードと低消費電力モードの少なくとも2つ以上の動作モードを持っている場合、スピンウェイト検出部は、スピンウェイト命令の実行中を検出した際にプロセッサの動作モードを低消費電力モードに変更する。また値変更検出部は、スピンウェイトの終了条件となっている変数値の変更を検出した際に、プロセッサの動作モードを通常モードに変更させる。これによりスピンウェイト実行中にプロセッサの消費電力量を削減できる。

15 プロセッサはプログラム実行を停止する機能を持っている場合、スピンウェイト検出部は、スピンウェイト命令の実行中を検出した際にプロセッサにプログラムの実行を停止させる。また値変更検出部は、スピンウェイトの終了条件となっている変数値の変更を検出した際に、プロセッサにプログラムの実行を再開させる。これによりスピンウェイト実行中にプログラムの実行が停止され、これに伴いプロセッサの消費電力量を削減できる。

20 プロセッサは、複数スレッドを同時実行できる構造（SMTプロセッサ）を持っている場合、スピンウェイト検出部は、スピンウェイト命令の実行中のスレッドを検出した際に、このスレッドへのプロセッサリソースの割当てを減少又は停止させる。また値変更検出部は、スピンウェイトの終了条件となっている変数値の変更を検出した際に、スレッドへのプロセッサリソースの割当てを増加又は再開させる。これによりSMPプロセッサにおけるスレッドのスピンウェイト実行中に、他のスレッドプログラムの実行が停止され、スピンウェイト実行スレッド

による他のスレッドへの悪影響を回避し、結果としてプロセッサの性能が上がる。

プロセッサはスピンウェイト専用命令を持っており、スピンウェイト検出部は、プロセッサによるスピンウェイト専用命令の実行からスピンウェイト実行中を検出する。またスピンウェイト検出部は、プロセッサによる実効済命令列を格納するバッファと、スピンウェイト専用命令の命令列を記録する命令列記録部と、バッファと命令列記録部とを比較して命令列が一致した際に、スピンウェイト実行中を検出する比較部とを備える。

本発明は、キャッシュメモリを備えた複数のプロセッサを有するコンピュータの制御方法を提供するものであり、

10 プロセッサによるスピンウェイト命令の実行中を検出した際に、スピンウェイトの終了条件となっている変数値の監視を開始した後にプロセッサの動作状態を変更させ、

監視中に変数値が変更されたことを検出した場合にプロセッサに値変更を通知して動作状態を元に戻させることを特徴とする。

15 この制御方法の詳細はコンピュータの場合と基本的に同じになる。

図面の簡単な説明

図1は従来のスピンロック処理の説明図；

図2は従来のバリア同期処理の説明図；

20 図3は本発明が適用される共有メモリ型マルチプロセッサシステムのブロック図；

図4は本発明によるプロセッサの実施形態のブロック図

図5は図4におけるスピンウェイト検出部の実施形態ブロック図；

図6はキャッシュブロックを監視する図4における値変更検出部の実施形態のブロック図；

25 図7はスピンウェイト検出時にキャッシュブロックを監視して値変更を検出する本発明の処理手順のタイムチャート；

図8は本発明による同期処理の説明図

図9はバストランザクションを監視する図4における値変更検出部の他の実施形態のブロック図；

図10は本発明によるスピンドルウェイト検出時にバストランザクションを監視して値変更を検出する本発明の処理手順のタイムチャート；

図11は本発明が適用されるSMTプロセッサの概略構成のブロック図；

図12はスピンドルウェイトを検出してプロセッサリソースを削減する本発明の処理

5 手順のタイムチャート；

発明を実施するための最良の形態

図3は、本発明が適用される共有メモリ型マルチプロセッサシステムのブロック図である。図3において、マルチプロセッサシステムは、例えば4台のプロセッサ10-1～10-4を備えており、プロセッサ10-1～10-4ごとにキャッシュメモリ12-1～12-4が設けられている。キャッシュメモリ12-1～12-4はバス14を介して共有メモリとして使用されるメインメモリ16を接続している。またバス14には入出力ユニット18が接続され、この入出力ユニット18としては例えば入出力エージェントとして機能する入出力プロセッサであり、外部のハードディスクドライブHDDなどのサブシステムやネットワーク、更にはキーボード、マウス、ディスプレイなどの入出力機器を接続している。

図4は、図3のプロセッサ10-1の実施形態である。なお、他のプロセッサ10-2～10-4も同じ実施形態となる。図4において、プロセッサ10-1は、プロセッサコア11、キャッシュメモリインタフェース20を備える。またキャッシュメモリ12-1は、プロセッサインタフェース24、キャッシュメモリ制御部26、バスインタフェース28及びキャッシュメモリアレイ30を備えている。このようなプロセッサ10-1及びキャッシュメモリ12-1の構成は従来のプロセッサと同じであるが、本発明にあってはプロセッサ10-1側に新たにスピンドルウェイト検出部22を設け、同時にキャッシュメモリ12-1側に値変更検出部32を設けている。プロセッサ10-1に設けられたスピンドルウェイト検出部22は、プロセッサコア11によるスピンドルウェイト命令の実行中を検出した際に、スピンドルウェイトの終了条件となっている変数値の監視をキャッシュメモリ12-1側に設けた値変更検出部32に指示した後に、プロセッサ10-1の

動作状態を変更させる。スピニウェイト検出部22によるプロセッサ10-1の動作状態の変更としては、スピニウェイト命令の実行中に生ずる無駄を削減する動作状態に変更するものであり、具体的には次のようになる。

例えばプロセッサ10-1が通常モードと低消費電力モードの少なくとも2つの動作モードを持っている場合には、スピニウェイト検出部22でスピニウェイト命令の実行中を検出した際にはプロセッサの動作モードを低消費電力モードに変更する。またプロセッサがプログラム実行を停止する機能を持っている場合には、スピニウェイト検出部22でスピニウェイト命令の実行中を検出した際にプロセッサにプログラムの実行を停止させる。このスピニウェイト命令の実行を検出した際の低消費電力モードへの変更あるいはプログラムの実行停止のいずれも、スピニウェイトの終了条件が所定の値に変更されるまで延々とスピニウェイトの命令列の実行を繰り返すスピニにより、プロセッサの無駄な動作や無駄に使用している消費電力を防ぐようとする。もちろん、スピニウェイト命令検出時のプロセッサによるプログラムの実行停止は低消費電力モードの1つの形態に含まれると言つこともできる。

キャッシュメモリ12-1側に設けた値変更検出部32は、スピニウェイト検出部22から指示されたスピニウェイトの終了条件となっている変数値を監視し、この変数値が他のプロセッサ10-2～10-4や入出力ユニット18により変更されたことを検出した場合に、プロセッサ10-1側に値変更を通知してプロセッサの動作状態を元に戻す。例えば、スピニウェイト命令実行中の検出でプロセッサが低消費電力モードに変更されていた場合には通常モードに戻す。またスピニウェイト命令の実行中の検出でプロセッサのプログラムの実行停止となつた場合にはプログラムの実行を再開させる。

図5は、図4のプロセッサ10-1に設けているスピニウェイト検出部22の実施形態である。スピニウェイト検出部22は、実行済命令列バッファ34、スピニウェイト命令列記録部36及び比較部38を備える。この実施形態にあっては、既存の命令セットISA(Instruction Set Architecture)をそのまま使用した場合を例に取つてゐる。スピニウェイト検出部22は、まずスピニウェイト命令列記録部36にスピニウェイトであることを示す命令列を予め記録する。この

命令列の記録は複数個でもよく、また記録内容は予め定められたものであっても良い。実行済命令列バッファ34には、プロセッサコアで実行が完了した命令列が記録される。比較部38は、スピニウェイト命令列記録部36に記録している基準となる命令列と実行済命令列バッファ34に記録された命令列を比較し、両
5 者が一致したら即ちスピニウェイト命令列が実行されていることを検出したら、キャッシュメモリインタフェース20に対しスピニウェイト命令列検出信号E1を出力する。

図6は、図4のキャッシュメモリ12-1側に設けている値変更検出部32の実施形態である。この実施形態において値変更検出部32には、監視対象先記録部40とキャッシュステート監視部42を設けている。プロセッサにおいてスピニウェイト命令の実行によりスピニウェイトが始まると、図5に示したスピニウェイト検出部22よりスピニウェイト命令列検出信号E1が出力され、図6の値変更検出部32に設けている監視対象アドレス記録部40にスピニウェイトの終了条件として参照している変数値を格納しているキャッシュブロック、具体的にはキャッシュメモリエントリを、キャッシュメモリ制御部26を経由して受けて記録する。キャッシュステート監視部42は監視対象アドレス記録部40に記録されたキャッシュメモリエントリを監視しており、このエントリのキャッシュステート（キャッシュ状態）が更新されたら、キャッシュメモリ制御部26を介してプロセッサ側に値変更検出信号E2を出力する。

20 この実施形態において、キャッシュプロトコルとしてはMESIプロトコルを例にとっており、この場合、キャッシュステート監視部42は監視対象となるエントリのキャッシュステートが無効Iに更新されたら値変更検出信号E2を出力する。なおキャッシュステート監視部42としては、スピニウェイト終了条件となっている変数値の書き換えに対応したキャッシュステートの無効Iの更新から値変更検出信号E2を出力しているが、キャッシュメモリのLRUアルゴリズム
25 により監視対象となっているキャッシュメモリエントリの追い出しが行われた場合にも値変更検出信号E2を出力する。

図7は、スピニウェイト検出時にスピニウェイト終了条件となっている変数値が格納されているキャッシュメモリエントリ（キャッシュブロック）を監視して

5 値変更を検出する本発明の処理手順のタイムチャートである。まずスピニウェイト検出部 2 2 がステップ S 1 でプロセッサ 1 0 - 1 でスピニウェイト命令を実行中か否かチェックしている。プロセッサ 1 0 - 1 でスピニウェイト命令が実行されると、図 5 の実施形態に示したようなスピニウェイト検出部 2 2 でスピニウェイト命令の実行が検出され、スピニウェイト命令検出信号 E 1 が出力される。これによって、ステップ S 2 でスピニウェイト検出部 2 2 は、スピニウェイトで参照している変数値が格納されたキャッシュブロックの監視をキャッシュメモリ 1 2 - 1 側の値変更検出部 3 2 に指示する。これと同時にスピニウェイト検出部 2 2 は、ステップ S 3 でプロセッサ 1 0 - 1 によるスピニウェイト命令の実行を停止し、低消費電力の動作状態に入る。

10 スピニウェイト検出部 2 2 からキャッシュブロックの監視を指示された値変更検出部 3 2 は、ステップ S 1 0 1 でプロセッサから指示されたキャッシュブロックの監視を開始する。即ち、ステップ S 1 0 2 で監視対象となるキャッシュブロックにおけるキャッシュステート即ちキャッシュ状態の状態変化、具体的には無効化が行われたか否かを検出している。この状態で他のプロセッサ 1 0 - 1 ~ 1 0 - 4 のいずれか、あるいは入出力ユニット 1 8 によるキャッシュブロックの更新があると、監視しているキャッシュブロックのキャッシュステートが無効状態に更新される。これをステップ S 1 0 2 で検出すると、ステップ S 1 0 3 に進み、キャッシュブロックの状態変化を図 6 の値変更検出部 3 0 より値変更検出信号 E 2 として出力し、キャッシュブロックの状態変化即ち無効化をプロセッサ側に通知する。この値変更検出信号を受けて、スピニウェイト検出部 2 2 にあっては、ステップ S 4 でプロセッサ 1 0 - 1 によるスピニウェイト命令の実行を再開する。

15 図 8 は、マルチプロセッサシステムにおけるバリア同期処理を例にとって本発明の処理手順を示した説明図である。図 8 (A) はバリア同期処理の対象となるマルチプロセッサシステムの一部であり、説明を簡単にするため 2 台のプロセッサ 1 0 - 1, 1 0 - 2 のバリア同期処理を例にとっている。またプロセッサ 1 0 - 1, 1 0 - 2 に設けたキャッシュメモリ 1 2 - 1, 1 2 - 2 にあっては、スピニウェイト命令の実行で参照しているスピニウェイトの終了条件となっている 2 ビットの変数値とキャッシュブロックにおけるキャッシュステートを示している。

このキャッシュステートはMESIプロトコルに対応しており、また変数値は2台のプロセッサ10-1, 10-2のバリア同期であることから、それぞれに対応した2ビットの情報であり、同期ポイントに達する前の状態では「00」となっている。

5 図8 (B) ~ (G) は、図7に示した本発明の処理手順に従ったバリア同期処理の動作の遷移状態を時間的に分けて表わしている。図8 (B) はバリア同期の対象となっている2台のプロセッサ10-1, 10-2の内、プロセッサ10-1が同期ポイントに達してスピニウェイトした状態である。即ちプロセッサ10-1は、同期ポイントへの到達で図2に示した設定部104の処理により、キャッシュメモリ12-1の対象エントリの変数値のうち自分の対応ビットを1にセットし、これにより変数値は「00」から「01」となる。このキャッシュメモリ12-1の変数値の更新に際し、同じキャッシュブロックを持っているプロセッサ10-2のキャッシュメモリ12-2にあっては、キャッシュステートの無効化が行われる。なおキャッシュメモリ12-1にあっては、該当するキャッシュブロックのキャッシュステートは変更Mに更新される。

この図8 (B) のようにプロセッサ10-1でスピニウェイト状態になると、図4のようにプロセッサ10-1に設けているスピニウェイト検出部22がスピニウェイト命令の実行中を検出し、キャッシュメモリ12-1に対し該当するキャッシュブロックのキャッシュステートが無効Iになるか否かの監視を指示する。20同時にプロセッサ10-1に対し例えばスピニウェイトのプログラム実行を停止させる。これにより図8 (C) のように、プロセッサ10-1はスピニウェイトのプログラム実行を停止した状態となり、一方、プロセッサ10-2は通常モードでプログラム実行を続けている。

図8 (D) はプロセッサ10-2が同期ポイントに達したときの処理である。25プロセッサ10-2が同期ポイントに達すると、スピニウェイト命令の実行によりキャッシュメモリ12-2からスピニウェイト終了条件となる変数値の読み込みを行う。このときキャッシュメモリ12-2の該当キャッシュブロックは無効I状態にあることから、キャッシュプロトコルによりプロセッサ10-1のキャッシュメモリ12-2から変数値を読み出してキャッシュステートを共有S状態

にした後にプロセッサ 10-2 で読み取る。続いて図 8 (E) のように、プロセッサ 10-2 は自分に割り当てられているビットを 1 に書き換えることで変数値を「11」とする。この変数値の更新により、プログラムの実行停止にあるプロセッサ 10-1 のキャッシュメモリ 12-1 に対し変数値の無効化が行われ、キャッシュステートはそれまでの共有 S から無効 I に更新される。このキャッシュメモリ 12-1 における無効 I への状態変化を図 4 のキャッシュメモリ 12-1 側に設けている値変更検出部 32 で検出すると、プロセッサ 10-1 に対し値変更通知を行い、これによって、それまで停止していたプログラムの実行が再開される。

このため、プログラムの実行を再開したプロセッサ 10-1 は、図 8 (F) でスピニウェイト命令の実行によりキャッシュメモリ 12-1 の該当キャッシュブロックの変数値「11」を読み込み、これによってスピニウェイト終了条件が成立したことを認識する。このときプロセッサ 12-2 側にあっても、変数値「11」の認識でスピニウェイト終了条件を判定しており、これにより 2 台のプロセッサ 10-1, 10-2 のバリア同期が取られ、図 8 (G) のようにスピニウェイトを終了して、次のプログラムの実行処理に移行することになる。

図 9 は、図 4 のキャッシュメモリ 12-1 側に設けている値変更検出部 32 の他の実施形態であり、この実施形態にあってはスピニウェイトの終了条件となっている変数値を格納しているキャッシュブロックに対する他のプロセッサ、あるいは入出力ユニットによる変数値変更のためのバストランザクション（バスリクエスト）を監視して値変更を検出するようにしたことを特徴とする。

図 9 の値変更検出部 32 は、図 6 と同じ監視対象アドレス記録部 40 に対しバストランザクション監視部 44 を設けている。プロセッサ側でスピニウェイト命令の実行が開始されると、スピニウェイトで参照しているデータアドレスが監視対象アドレス記録部 40 に記録される。バストランザクション監視部 44 は、監視対象アドレスに対する他のプロセッサあるいは入出力ユニットからのバス 14 によるトランザクションをバスインターフェース 28 を介して監視しており、監視しているバストランザクションを検出すると、値変更検出信号 E2 をキャッシュメモリ制御部 26 を経由してプロセッサ側に出力する。このバストランザクシ

ヨンの監視による値変更検出は、スピニウェイトの終了条件となっている変数値の変更を直接検出するものではなく、正確には変数値が変更される可能性があることを検出していると言うことができる。

図10は、図9のバストランザクションの監視により値変更を検出する場合の5 本発明の処理手順のタイムチャートである。スピニウェイト検出部2.2は、ステップS1でプロセッサでのスピニウェイト命令実行中を検出すると、ステップS2で、スピニウェイト命令で参照している変数値が格納されているキャッシュブロック、即ちデータアドレスの監視の指示をキャッシュメモリ側の値変更検出部3.2に対し行う。またステップS3で、スピニウェイト検出部2.2はプロセッサ10 に対しスピニウェイト命令列検出実行を出力して、プロセッサによるスピニウェイト命令の実行を停止させる。

キャッシュメモリ側の値変更検出部3.2にあっては、ステップS10.1でプロセッサ側から指示されたデータアドレスに対するバストランザクションを監視し、監視対象のキャッシュブロックに対するバストランザクションをステップS10.2で検出すると、ステップS10.3でバストランザクションによる状態変化（可能性）を値変更検出信号の出力によりプロセッサ側に通知する。これを受けてスピニウェイト検出部2.2にあっては、ステップS4でプロセッサによるスピニウェイト命令の実行を再開することになる。

次に図3のプロセッサ10-1～10-4としてSMTプロセッサを使用した20 場合について、スピニウェイト命令実行中のプロセッサに対するプロセッサリソースの割当てを削減する本発明の他の実施形態を説明する。

図11は、本発明が適用されるSMTプロセッサの概略構成である。SMTプロセッサは、命令ポインタ46-1, 46-2、フェッチ/デコード48-1, 48-2、実行ユニット50、リタイヤメントユニット52、1次命令キャッシュユニット54、1次データキャッシュユニット56及び2次キャッシュユニット58を備えている。このような概略構成を持つSMTプロセッサは、同時に複数スレッドを実行するため通常は命令ポインタを複数持つおり、この例では2つの命令ポインタ46-1, 46-2を持った場合を示している。この命令ポインタ46-1, 46-2及びフェッチ/デコード48-1, 48-2で実現され

る2つのスレッドの実行にあっては、実行ユニット50などのリソースをスレッド間で共有することにより、リソースの有効利用率を高めており、これがSMTプロセッサの狙いである。このように複数のスレッドでプロセッサリソースを共有するSMTプロセッサにあっては、次にどのスレッドの命令を実行ユニット50に割り当てるかを極める方針がいろいろ提案されている。そこで本発明は、この実行ユニットなどのプロセッサリソースをどのスレッドの命令に割り当てるかを決める方針の決定に利用することができる。具体的には、スピニウェイト実行中のスレッドを検出したら、スピニウェイト命令を実行中のスレッドに対する実行ユニットなどのプロセッサリソースの割当てを削減もしくは停止する。これによってスピニウェイトではない、より意味のある命令を実行している他のスレッドの命令を実行ユニットに割り当てる機会が増え、その結果、SMTプロセッサの処理性能の向上が期待できる。

図12は、SMTプロセッサについてスピニウェイトを検出してプロセッサリソースを削減する本発明の処理手順のタイムチャートである。SMTプロセッサ側のスピニウェイト検出部22は、ステップS1で複数のスレッドの内のあるスレッドでスピニウェイト実行中か否かをチェックしており、実行中を検出すると、ステップS2に進み、スピニウェイトで参照している変数値が格納されたキャッシュブロックの監視をキャッシュメモリ側の値変更検出部32に指示する。またステップS3で、スピニウェイト命令を実行中のスレッドの命令を割り当てるプロセッサリソースの削減または停止を行う。具体的には、例えば命令ポインタ46-1及びフェッチ/デコード48-1側で実行されるスレッドの命令でスピニウェイト実行中が検出された場合には、このスレッドによる命令の実行ユニット50に対する割当てを削減または停止し、スピニウェイト命令以外の命令を実行している命令ポインタ46-2、フェッチ/デコード48-2側となる他のスレッドの命令に実行ユニット50を割り当てる。

キャッシュメモリ側の値変更検出部32にあっては、ステップS101でプロセッサ側から指示されたキャッシュブロックのキャッシュステートを監視し、ステップS102でキャッシュステート即ち状態が無効Iに更新されたことを検出すると、ステップS103でキャッシュブロックの状態変化をプロセッサ側に通

知する。これを受けスピニウェイト検出部22側にあっては、ステップS4でスピニウェイト命令を実行中のスレッドに対するそれまでのプロセッサリソースの割当ての削減または停止を解除する。

なお上記の実施形態にあっては、キャッシングプロトコルとしてMESIプロトコルを例にとるものであったが、無効Iへの更新を含むものであれば適宜のキャッシングプロトコルであってもよい。また上記の実施形態にあっては、スピニウェイト終了条件のための変更値として該当するキャッシングロックのキャッシングステートの無効化、あるいはこのステートに対するバストランザクションからキャッシング終了条件となっている変数値を直接または間接的に監視しているが、これ以外の変数値の直接または間接的な監視であっても良いことはもちろんである。

また上記の実施形態にあっては、スピニウェイト実行中を検出した際のプロセッサの動作状態の変更として、低消費電力モードへの変更、プログラムの実行停止、更にはSMTプロセッサにあってはプロセッサリソースの割当て減少または停止を例に取るものであったが、これ以外に適宜のスピニウェイト命令の実行で生ずる無駄を省くプロセッサの動作状態の変更であれば適宜のものを含む。

また本発明は上記の実施形態に限定されず、その利点と目的を損なうことのない適宜の変形を含む。更に本発明は上記の実施形態に示した数値による限定は受けない。

20 【産業上の利用可能性】

以上説明してきたように本発明によれば、スピニウェイトの特徴である遅延の低さを犠牲にすることなく、スピニウェイト命令実行によって発生する無駄な命令実行やプロセッサの動作を制限することができ、その結果としてプロセッサの電力消費を節減することができる。

25 またSMTプロセッサにあっては、スピニウェイト実行スレッドによる他のスレッドへの悪影響を回避することができ、SMTプロセッサの処理能力を向上させることができる。

請求の範囲

1. キャッシュメモリを備えた複数のプロセッサを有するコンピュータに於いて、

前記プロセッサによる спинウェイト命令の実行中を検出した際に、前記スピ

5 ネウェイトの終了条件となっている変数値の監視を指示した後に前記プロセッサ
の動作状態を変更させるスピニウェイト検出部と、

前記スピニウェイト検出部から指示された変数値を監視し、該変数値が変更さ
れたことを検出した場合に前記プロセッサに値変更を通知して動作状態を元に戻
す値変更検出部と、

10 を備えたことを特徴とするコンピュータ。

2. 請求の範囲 1 のコンピュータにおいて、前記スピニウェイト検出部は、前記

プロセッサによるスピニウェイト命令の実行中を検出した際に、前記スピニウェ

イト命令の実行で参照している前記変数値が格納された前記キャッシュメモリの
キャッシュブロックの監視を前記値変更検出部に指示し、

前記値変更検出部は、前記スピニウェイト検出部から指示されたキャッシュブ
ロックを監視し、該キャッシュブロックの状態が変更されたことを検出した場合
に前記プロセッサに値変更を通知することを特徴とするコンピュータ。

20 3. 請求の範囲 2 のコンピュータに於いて、前記値変更検出部は、前記キャッシュ
ブロックの状態が他のプロセッサ又は入出力ユニットにより無効化されたこと
を検出した場合に、前記プロセッサに値変更を通知することを特徴とするコンピ
ュータ。

25 4. 請求の範囲 2 のコンピュータに於いて、前記値変更検出部は、前記キャッシュ
ブロックがキャッシュメモリから追い出されたことを検出した場合に、前記プロ
セッサに値変更を通知することを特徴とするコンピュータ。

5. 請求の範囲 1 のコンピュータに於いて、

前記スピニウェイト検出部は、前記プロセッサによるスピニウェイト命令の実行中を検出した際に、前記スピニウェイト命令の実行で参照している変数値が格納されているメモリアドレスに対するバスリクエストの監視を前記値変更検出部に指示し、

5 前記値変更検出部は、前記スピニウェイト検出部から指示された変数値が格納されているメモリアドレスに対するバスリクエストを監視し、他のプロセッサ又は入出力ユニットから前記メモリアドレスに対するバスリクエストを検出した場合に、前記プロセッサに値変更を通知することを特徴とするコンピュータ。

10 6. 請求の範囲 1 のコンピュータに於いて、

前記プロセッサは通常モードと低消費電力モードの少なくとも 2 つ以上の動作モードを持っており、

前記スピニウェイト検出部は、スピニウェイト命令の実行中を検出した際に前記プロセッサの動作モードを低消費電力モードに変更し、

15 前記値変更検出部は、前記スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記プロセッサの動作モードを通常モードに変更させることを特徴とするコンピュータ。

7. 請求の範囲 1 のコンピュータに於いて、

20 前記プロセッサはプログラム実行を停止する機能を持っており、

前記スピニウェイト検出部は、スピニウェイト命令の実行中を検出した際に前記プロセッサにプログラムの実行を停止させ、

前記値変更検出部は、前記スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記プロセッサにプログラムの実行を再開させることを特徴とするコンピュータ。

8. 請求の範囲 1 のコンピュータに於いて、

前記プロセッサは、複数スレッドを同時実行できる構造を持っており、

前記スピニウェイト検出部は、スピニウェイト命令の実行中のスレッドを検出

した際に、該スレッドへのプロセッサリソースの割当てを減少又は停止させ、

前記値変更検出部は、前記スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記スレッドへのプロセッサリソースの割当てを増加又は再開させることを特徴とするコンピュータ。

5

9. 請求の範囲 1 のコンピュータに於いて、前記プロセッサはスピニウェイト専用命令を持っており、前記スピニウェイト検出部は、前記プロセッサによるスピニウェイト専用命令の実行からスピニウェイト実行中を検出することを特徴とするコンピュータ。

10

10. 請求の範囲 1 のコンピュータに於いて、前記スピニウェイト検出部は、前記プロセッサによる実効済命令列を格納するバッファと、

前記スピニウェイト専用命令の命令列を記録するスピニウェイト命令列記録部と、

15 前記バッファとスピニウェイト命令列記録部とを比較して命令列が一致した際に、スピニウェイト実行中を検出することを特徴とするコンピュータ。

11. キャッシュメモリを備えた複数のプロセッサを有するコンピュータの制御方法に於いて、

20 前記プロセッサによるスピニウェイト命令の実行中を検出した際に、前記スピニウェイトの終了条件となっている変数値の監視を開始した後に前記プロセッサの動作状態を変更させ、

前記変数値が変更されたことを検出した場合に前記プロセッサに値変更を通知して動作状態を元に戻すことを特徴とするコンピュータの制御方法。

25

12. 請求の範囲 11 のコンピュータの制御方法に於いて、

前記プロセッサによるスピニウェイト命令の実行中を検出した際に、前記スピニウェイト命令の実行で参照している前記変数値が格納された前記キャッシュメモリのキャッシュブロックの監視を開始し、

前記キャッシュブロックの監視中に、該キャッシュブロックの状態が変更されたことを検出した場合に前記プロセッサに値変更を通知することを特徴とするコンピュータの制御方法。

5 13. 請求の範囲 1 2 のコンピュータの制御方法に於いて、前記キャッシュブロックの状態が他のプロセッサ又は入出力ユニットにより無効化されたことを検出した場合に、前記プロセッサに値変更を通知することを特徴とするコンピュータの制御方法。

10 14. 請求の範囲 1 2 のコンピュータの制御方法に於いて、前記キャッシュブロックがキャッシュメモリから追い出されたことを検出した場合に、前記プロセッサに値変更を通知することを特徴とするコンピュータの制御方法。

15. 請求の範囲 1 1 のコンピュータの制御方法に於いて、

15 前記プロセッサによるスピニウェイト命令の実行中を検出した際に、前記スピニウェイト命令の実行で参照している変数値が格納されているメモリアドレスに対するバスリクエストの監視を開始し、

前記メモリアドレスに対するバスリクエストの監視中に、他のプロセッサ又は入出力ユニットから前記メモリアドレスに対するバスリクエストを検出した場合に、前記プロセッサに値変更を通知することを特徴とするコンピュータの制御方法。

16. 請求の範囲 1 1 のコンピュータの制御方法に於いて、

前記プロセッサは通常モードと低消費電力モードの少なくとも 2 つ以上の動作モードを持っており、

前記スピニウェイト命令の実行中を検出した際に前記プロセッサの動作モードを低消費電力モードに変更させ、

前記スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記プロセッサの動作モードを通常モードに変更させることを特徴とするコンピュ

ータの制御方法。

17. 請求の範囲 11 のコンピュータの制御方法に於いて、

前記プロセッサはプログラム実行を停止する機能を持っており、

5 前記スピニウェイト命令の実行中を検出した際に前記プロセッサにプログラムの実行を停止させ、

前記スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記プロセッサにプログラムの実行を再開させることを特徴とするコンピュータの制御方法。

10

18. 請求の範囲 11 のコンピュータの制御方法に於いて、

前記プロセッサは、複数スレッドを同時実行できる構造を持っており、

前記スピニウェイト命令の実行中のスレッドを検出した際に、該スレッドへのプロセッサリソースの割当てを減少又は停止させ、

15 スピニウェイトの終了条件となっている変数値の変更を検出した際に、前記スレッドへのプロセッサリソースの割当てを増加又は再開させることを特徴とするコンピュータの制御方法。

19. 請求の範囲 11 のコンピュータの制御方法に於いて、前記プロセッサはスピニウェイト専用命令を持っており、前記プロセッサによるスピニウェイト専用命令の実行からスピニウェイト実行中を検出することを特徴とするコンピュータの制御方法。

20. 請求の範囲 19 のコンピュータの制御方法に於いて、前記スピニウェイト検出は、

前記スピニウェイト専用命令の命令列をスピニウェイト命令列記録部に記録し、前記プロセッサによる実行済命令列を順次バッファに格納し、

前記バッファとスピニウェイト命令列記録部とを比較して命令列が一致した際に、スピニウェイト実行中を検出することを特徴とするコンピュータの制御方法。

図 1

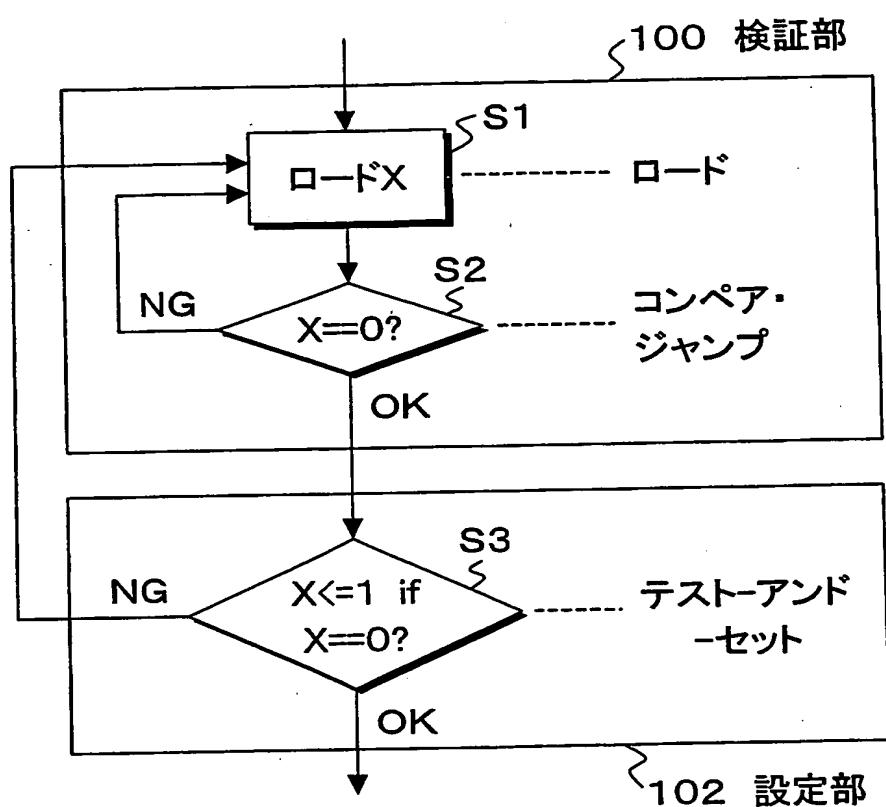
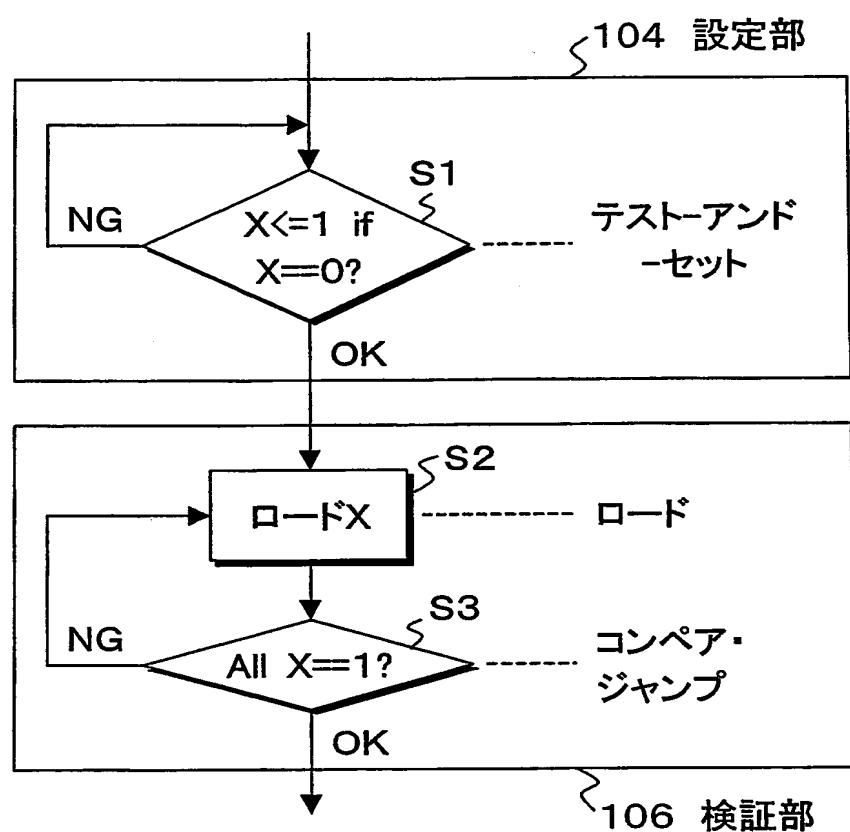


図2



3/12

図3

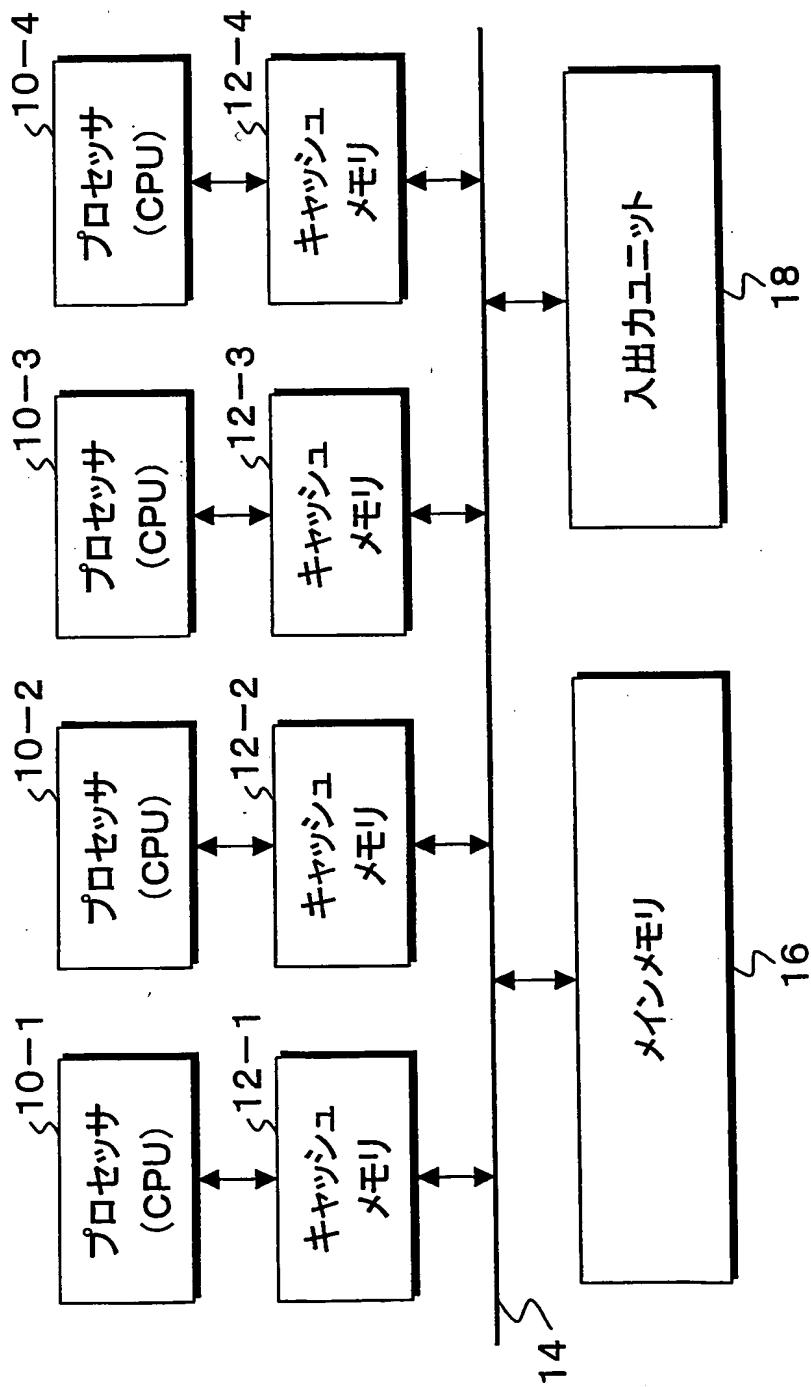
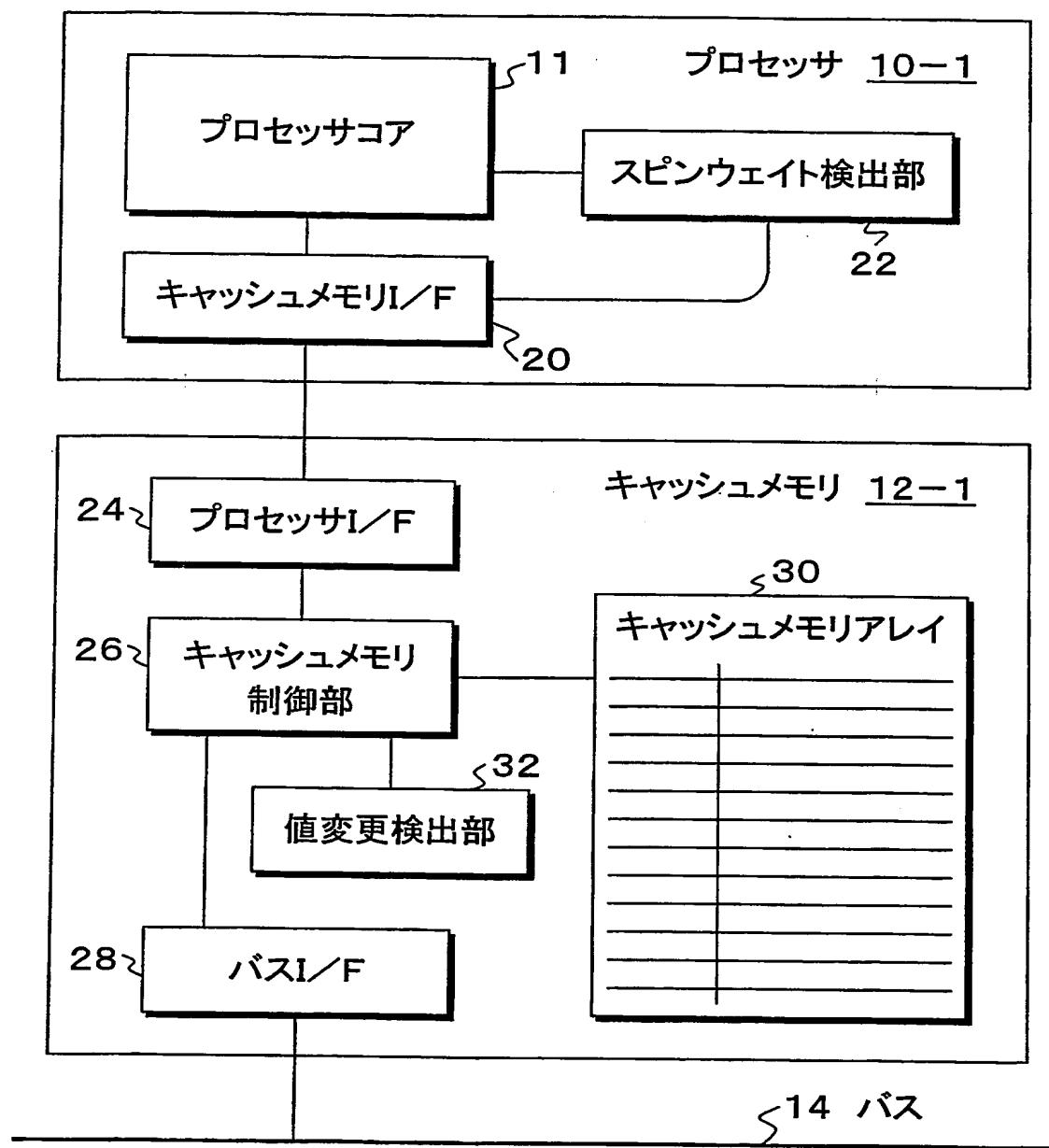
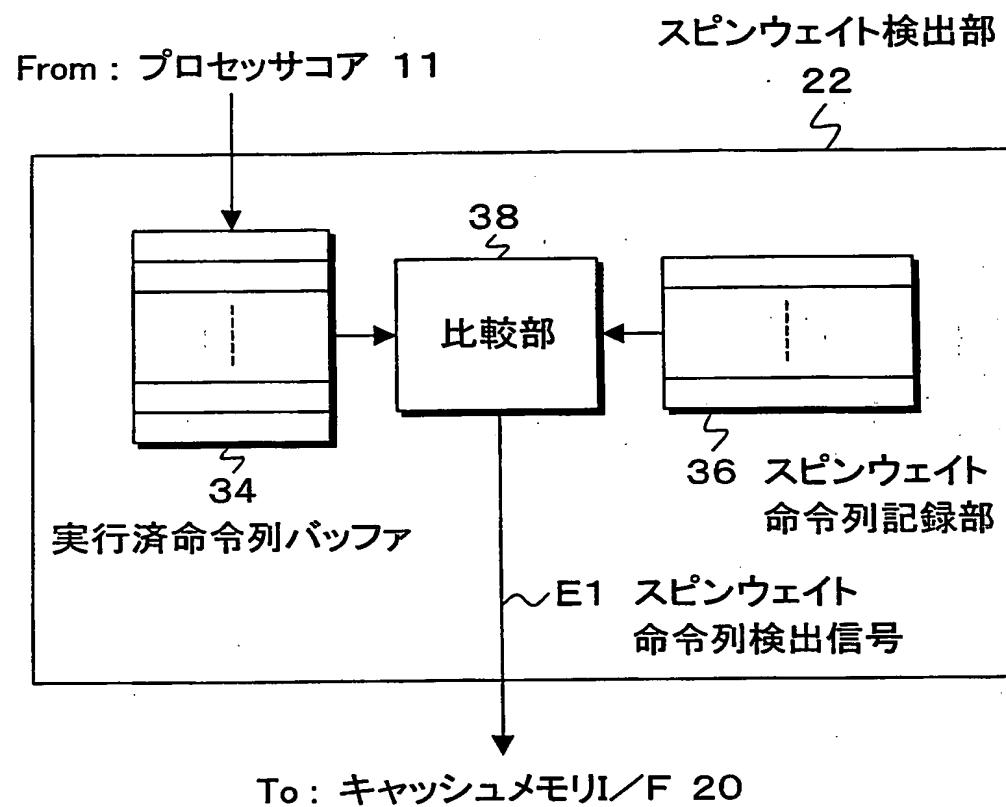


図4



5/12

図5



6/12

図6

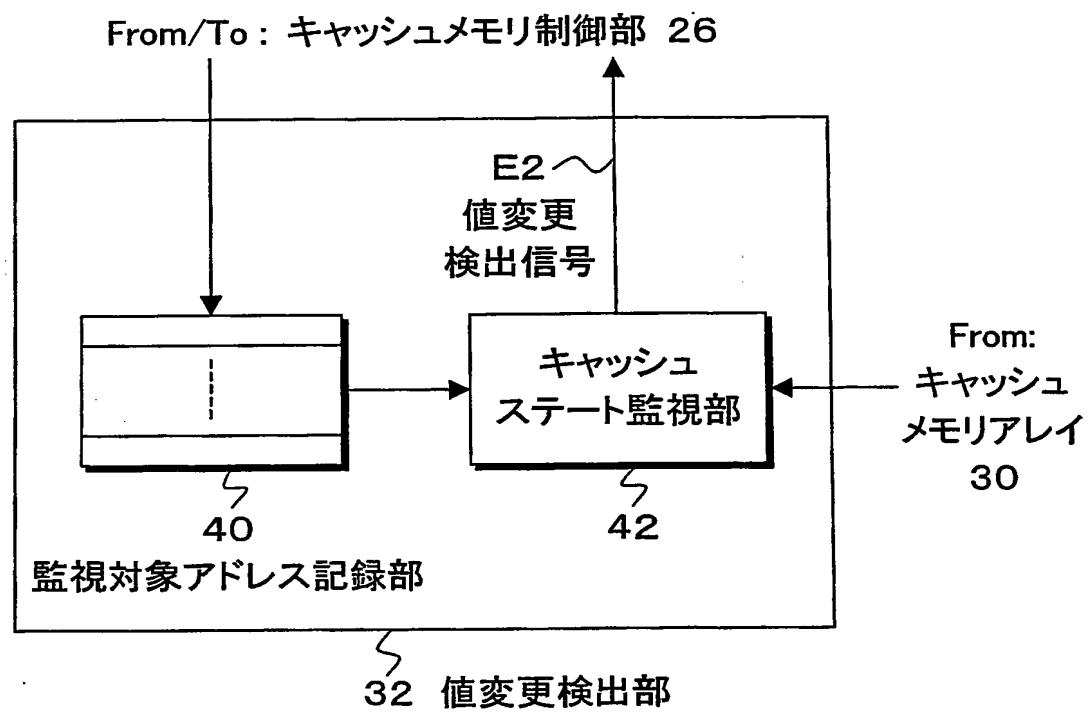
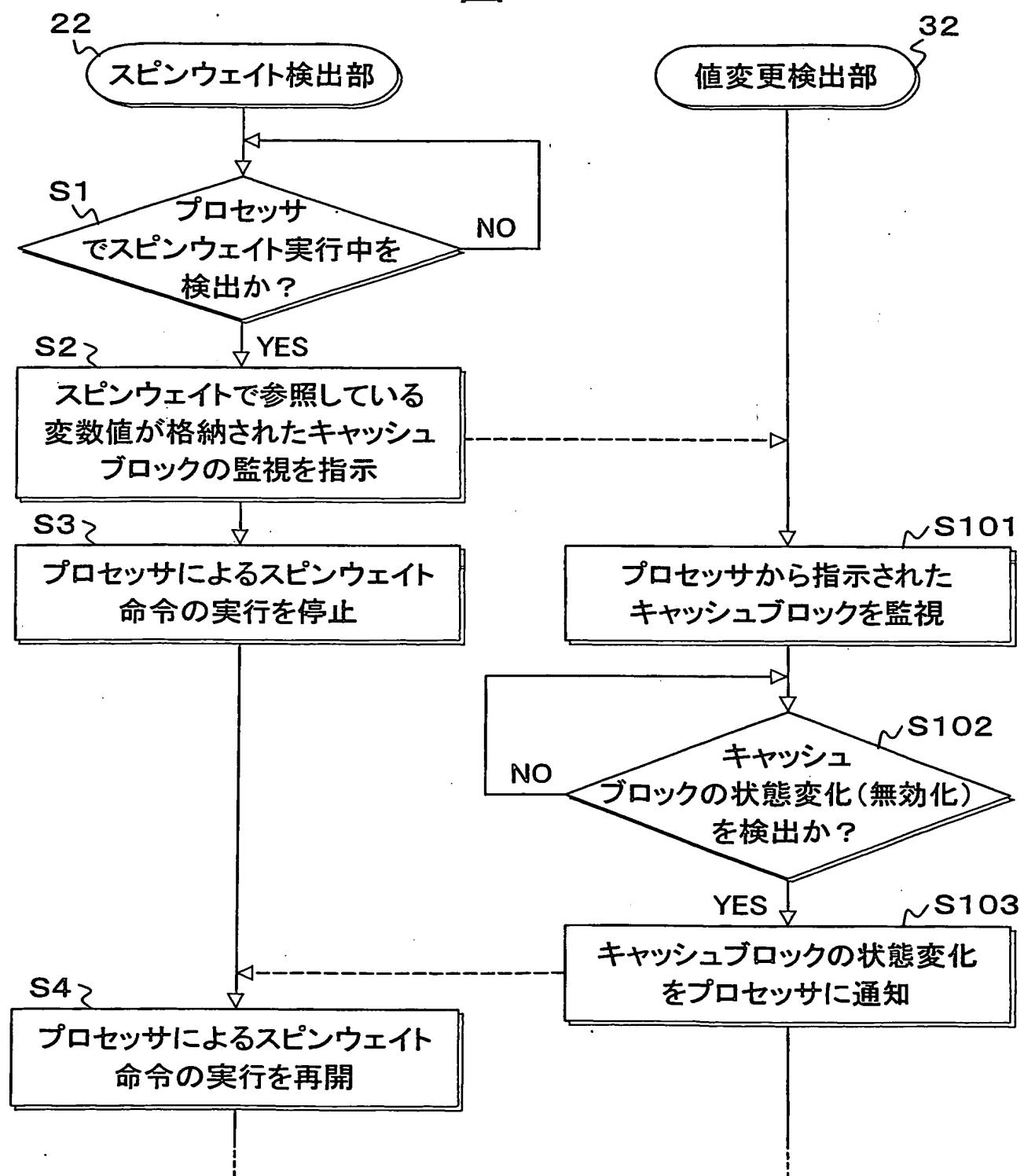
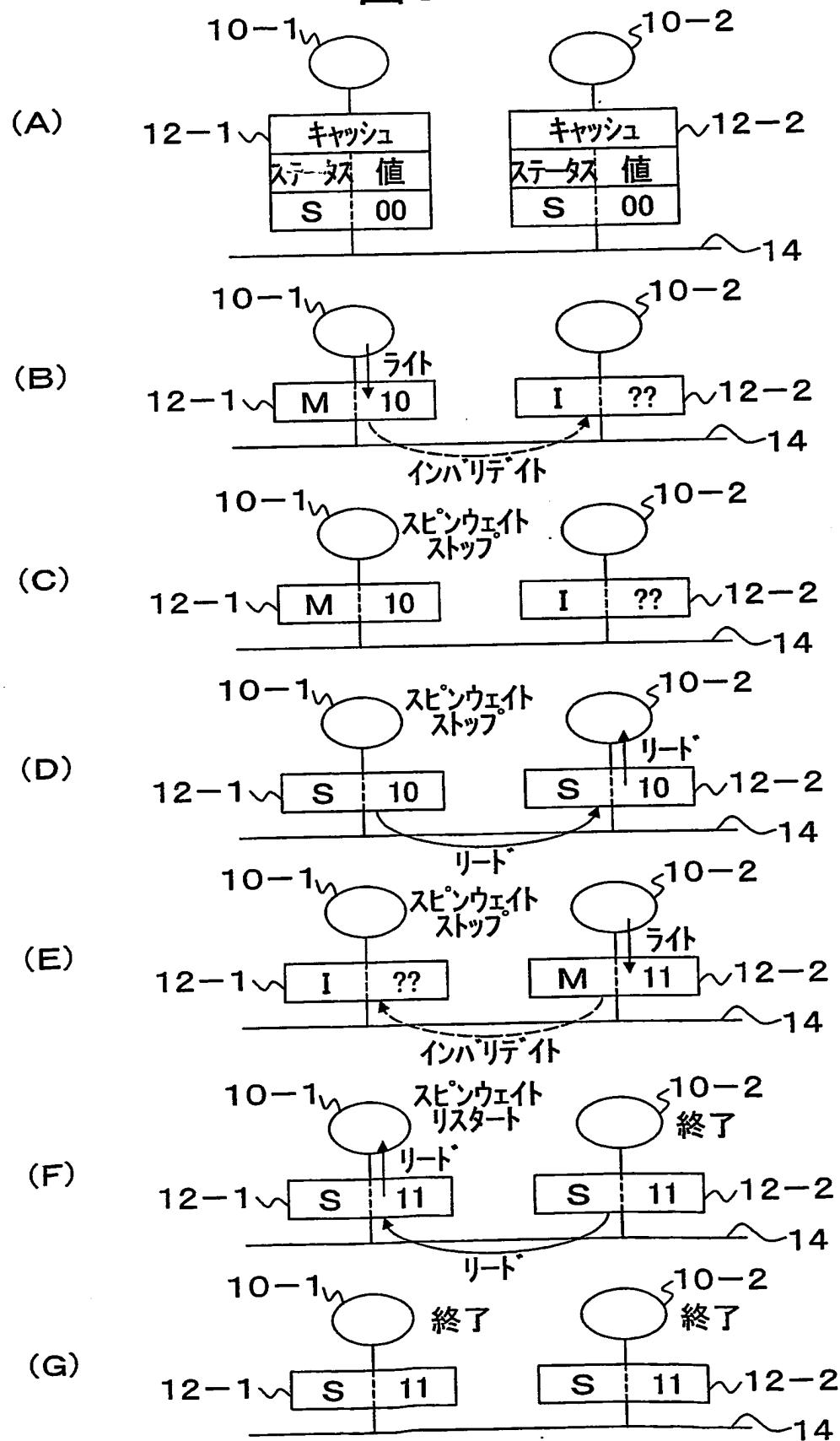


図7



8/12

図8



9/12

図9

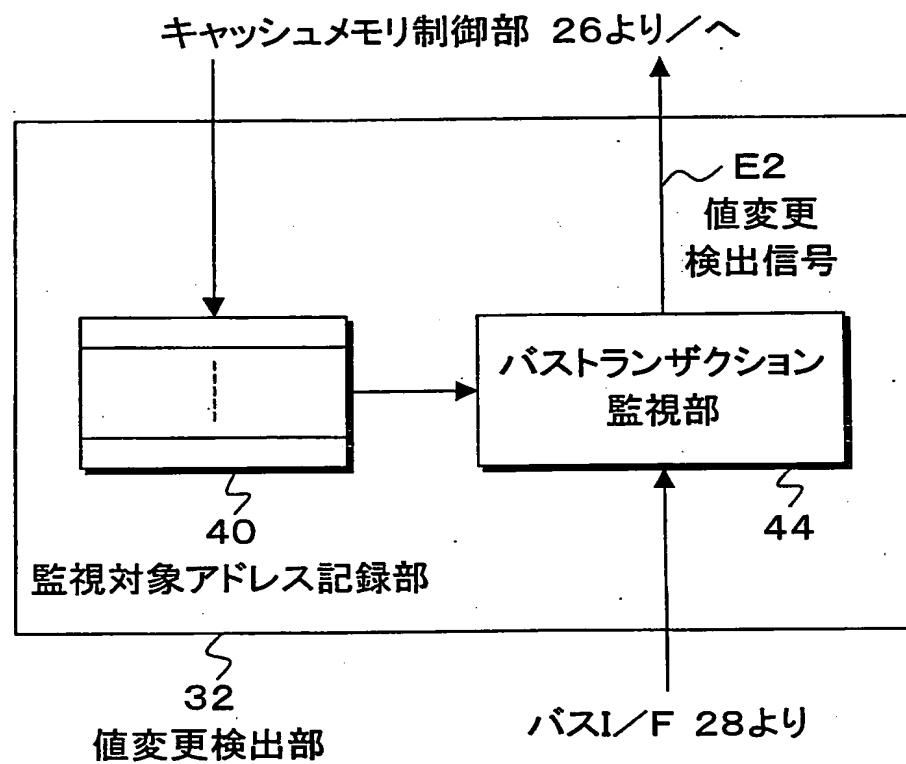


図10

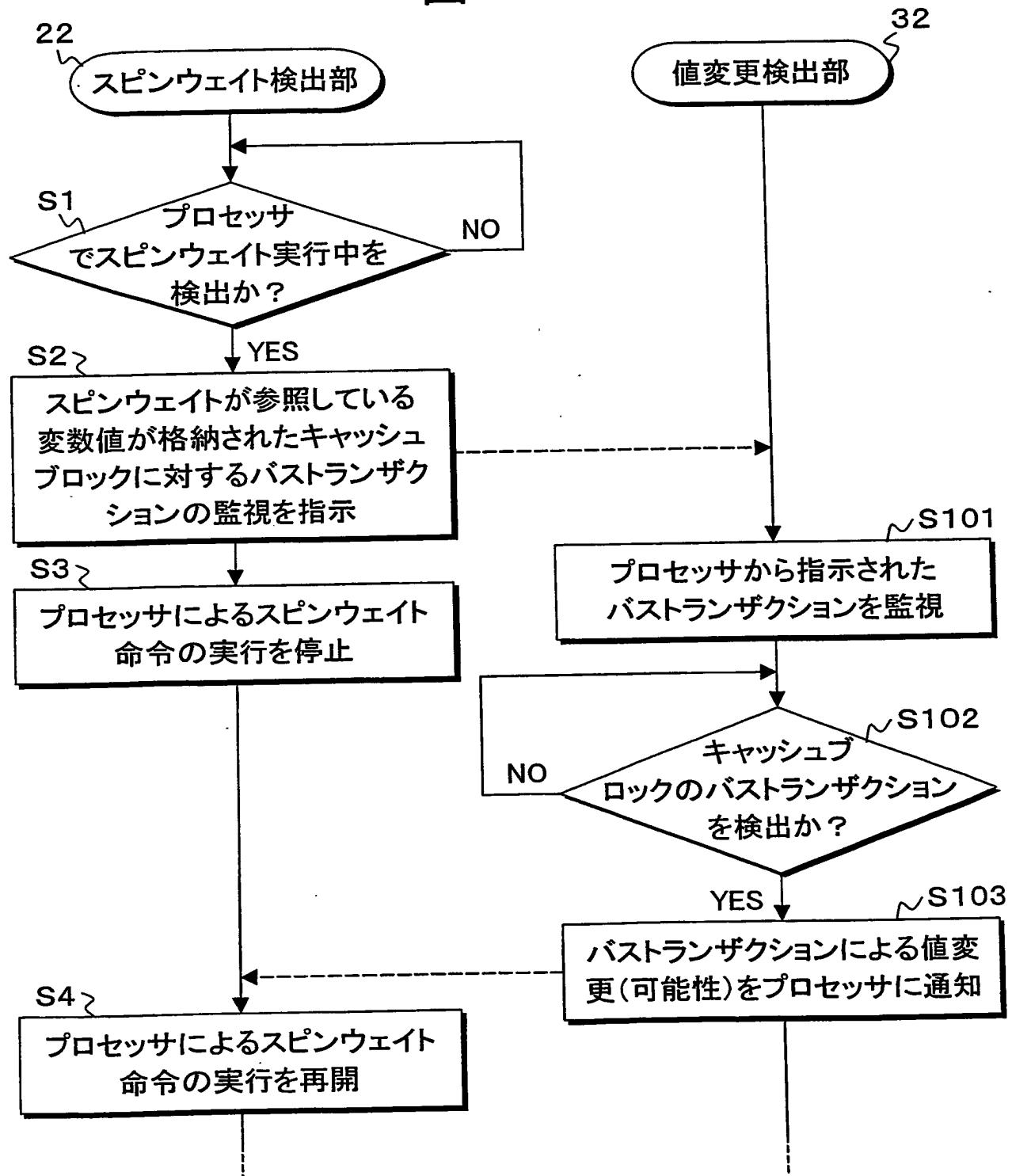
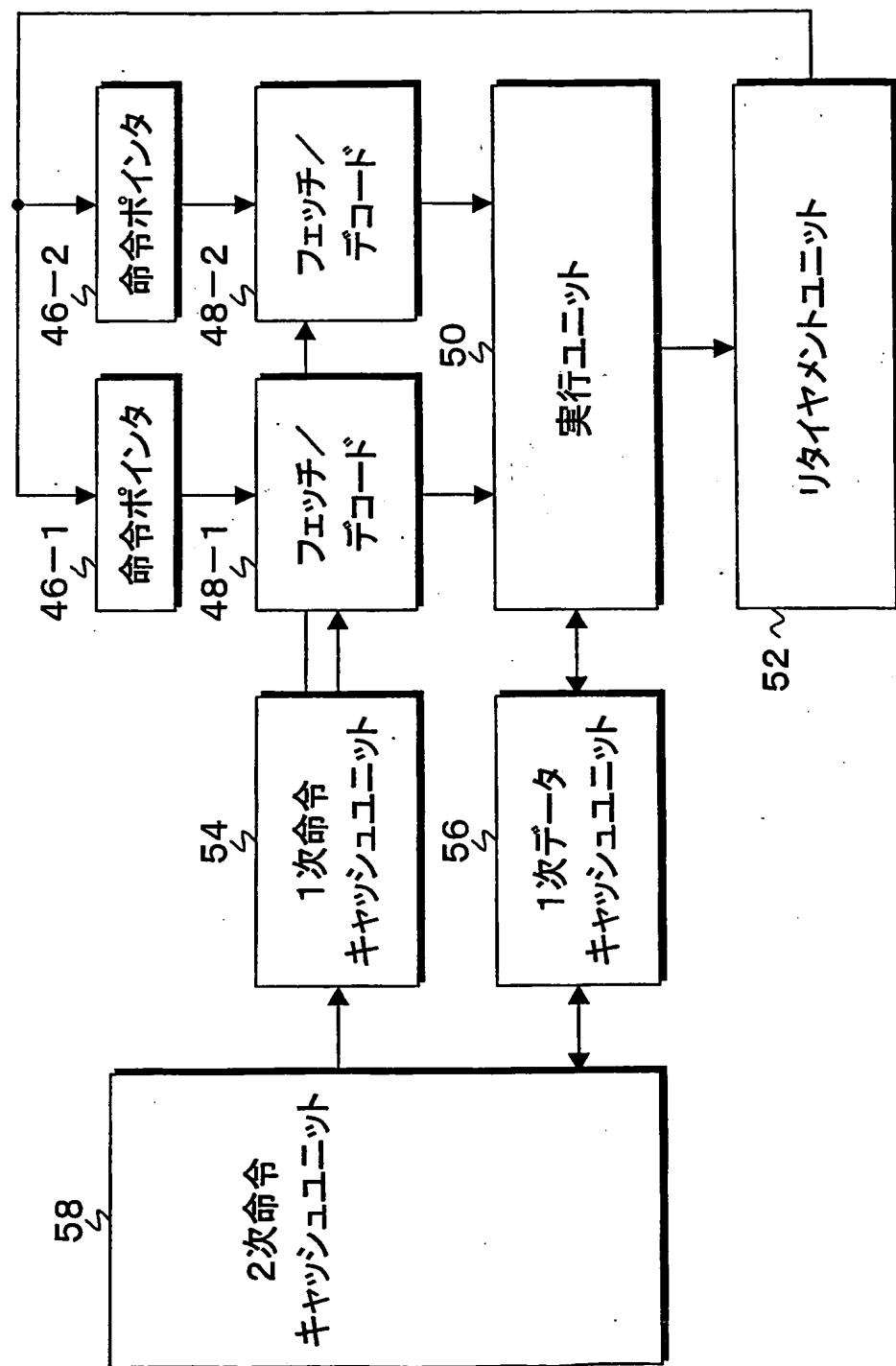
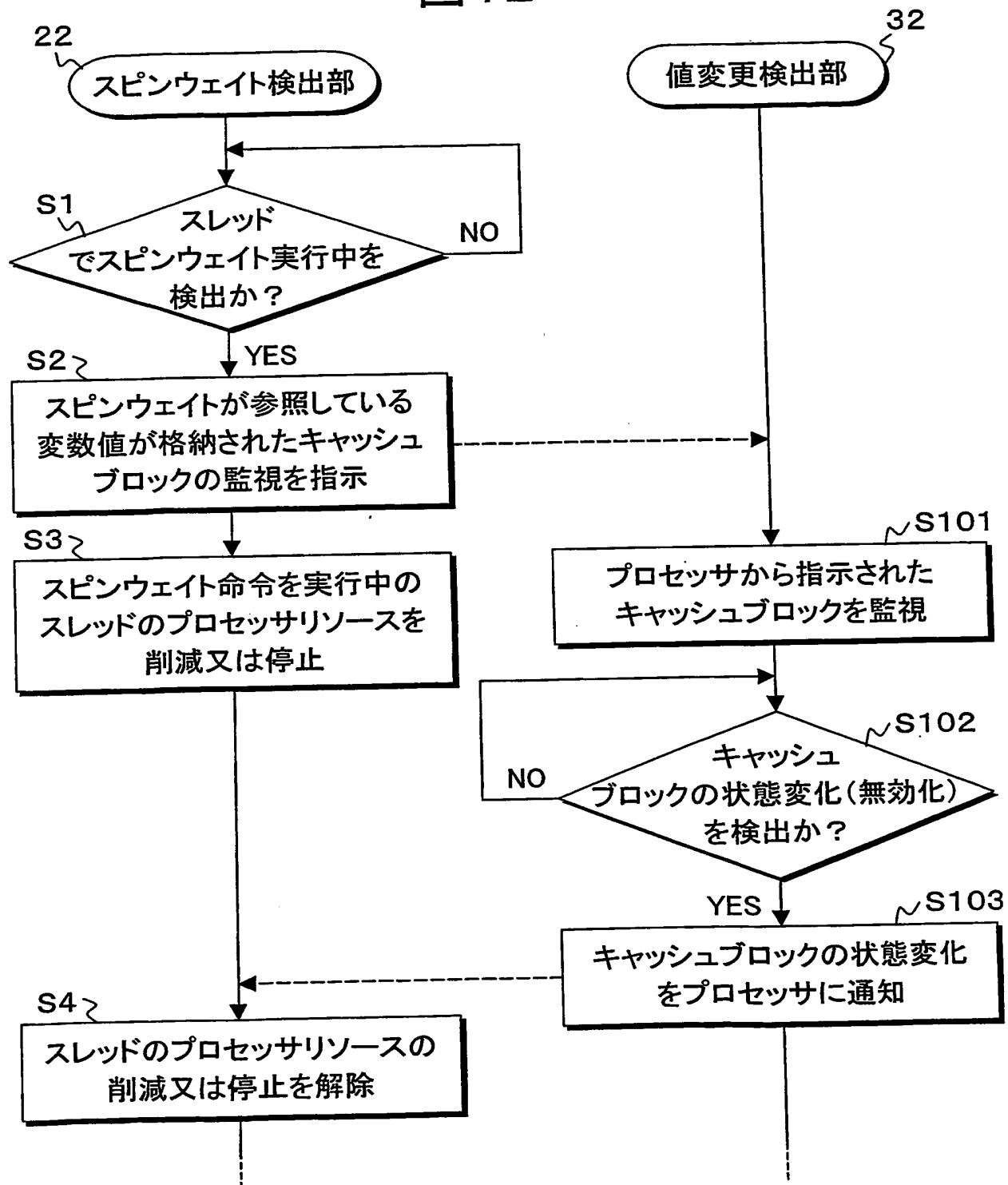


図 11



12/12

図12



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP01/09788

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl' G06F15/16

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl' G06F15/16, G06F12/08Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Jitsuyo Shinan Koho 1926-1996 Jitsuyo Shinan Toroku Koho 1996-2002
Kokai Jitsuyo Shinan Koho 1971-2002 Toroku Jitsuyo Shinan Koho 1994-2002

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Using Spin-Loops on Intel Pentium 4 Processor and Intel Xeon Processor, [Online], Intel Corp., May 2001, [retrieved on 2002-2-08], Retrieved from the Internet: <URL:ftp://download.intel.com/design/perftool/cbts/appnotes/sse2/w_spinlock.pdf>.	1,6-9, 11,16-19
A	ibid.	2-5,10, 12-15,20

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document but published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means		
"P" document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search
12 February, 2002 (12.02.02)Date of mailing of the international search report
26 February, 2002 (26.02.02)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int Cl' G 06 F 15/16

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int Cl' G 06 F 15/16, G 06 F 12/08

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1926-1996年
 日本国公開実用新案公報 1971-2002年
 日本国実用新案登録公報 1996-2002年
 日本国登録実用新案公報 1994-2002年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	Using Spin-Loops on Intel Pentium 4 Processor and Intel Xeon Processor, [Online], Intel Corp., May 2001, [retrieved on 2002-2-08], Retrieved from the Internet:<URL:ftp://download.intel.com/design/perftool/cbts/appnotes/sse2/w_spinlock.pdf>.	1, 6-9, 11, 16-19
A	ibid.	2-5, 10, 12-15, 20

 C欄の続きにも文献が列挙されている。 パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

- 「A」特に関連のある文献ではなく、一般的技術水準を示すもの
- 「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
- 「I」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)
- 「O」口頭による開示、使用、展示等に言及する文献
- 「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

- 「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
- 「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
- 「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
- 「&」同一パテントファミリー文献

国際調査を完了した日

12. 02. 02

国際調査報告の発送日

26.02.02

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号 100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

川崎 優

5B 8944



電話番号 03-3581-1101 内線 3545